Collaboratively developing workflows at the BSC-ES

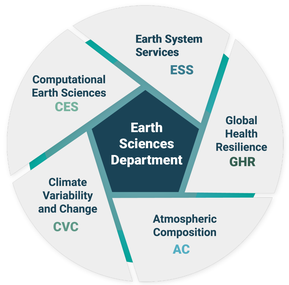
Gilbert Montané Pinto, Eric Ferrer, Miriam Olid, Alejandro Garcia, Genís Bonet, Amirpasha Mozaffari

Earth Sciences Department, Barcelona Supercomputing Center, Spain (gilbert.montane@bsc.es)

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

FOCI
Co-funded by the European Union

Introduction: the BSC-ES department



The Earth Sciences department of the Barcelona Supercomputing Center (BSC-ES) is one of the most active groups in research and operational modelling of air quality and atmospheric composition, climate prediction and climate services in Europe. It currently consists of five research groups and about 180 researchers and research engineers, including also support staff.

Being such a large department with different profiles working on different research topics implies running a variety of workflows for many different purposes, such as executing weather and climate simulations, downloading data or evaluating model performance. This means having to deal with many different processes and time scales in different environments and machines.

The Autosubmit workflow manager

To support complex tasks like running weather and climate experiments, the Autosubmit workflow manager has been developed to be used by the whole department as a unique framework to carry out Earth science-related simulations, mainly in HPC environments.

Autosubmit is a Python-based tool to create, manage and monitor many types of experiments remotely without manual intervention. It can orchestrate all the tasks in a workflow by managing their dependencies, interfacing with all the platforms involved, and handling eventual errors.

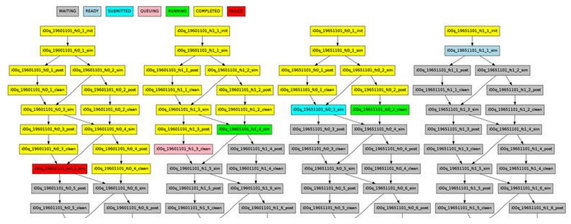
AUTOSUBMIT

ES workflows with Autosubmit

In this context we can define an Earth Science modeling workflow as the set of individual tasks that are needed to perform a simulation with a weather or a climate model. Tasks may have dependencies between them, i.e. a task can start only when other (parent) tasks have finished successfully. Each task may run in a different platform, usually an HPC system. All these facts increase the complexity of this kind of workflow.

A weather or climate simulation can be divided in different phases, which largely determine the dependencies between the jobs in a workflow:

In Autosubmit, a complex workflow may look like this:

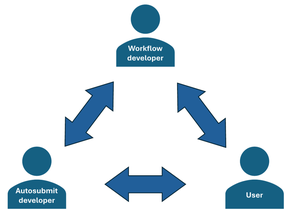


Example of a complex workflow created with Autosubmit. Each box is a job, i.e. an individual step in the workflow. The arrows represent the dependencies between jobs and the colors represent the current status of each job.

Workflow development at the BSC-ES

The fact that Autosubmit has been fully developed at BSC-ES has led to the creation of strong synergies between department users (scientists), workflow developers and Autosubmit developers, and to the adoption of a co-design procedure to fulfill all the objectives, which has contributed to a significant increase in scientific productivity.

- **Workflow developers**: gather specific user requirements, design and develop workflows, and communicate with Autosubmit developers to implement new features if required.
- **Autosubmit developers**: receive feedback and resolve issues reported by both workflow engineers and users, designing and implementing new features as needed.
- **Users**: report any issue or request they might have to Autosubmit and workflow developers.



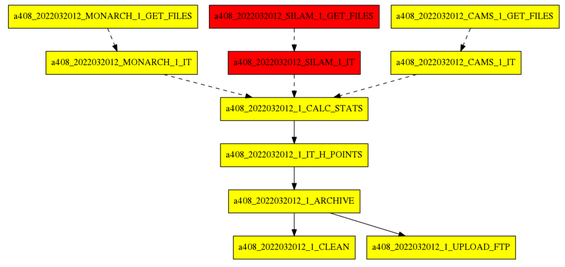
Workflow developers try to meet user requirements as they arise. If this is not possible with the options currently available, the Autosubmit developers will implement a new feature so that everyone can benefit. The following sections describe some examples of features that were implemented from particular use cases.

Feature example 1: Weak dependencies

In some operational forecast executions, the results have to be delivered at a given time using the available input data at that time, even if not totally complete (initial and boundary conditions, observations, etc.).

The aim of the weak dependencies feature is to allow the execution of a particular task even if any (not all) of its parent tasks have failed.

As an example, we could have a workflow that runs an ensemble of different air quality models. It could happen that one or more model outputs are missing when the workflow is executed, but anyway we need to run the ensemble with the available ones. The following figure shows an example of how this type of workflow will look in Autosubmit:



Example of ensemble workflow using weak dependencies (dashed arrows). Each box represent a task (a single step in the workflow), arrows represent dependencies between jobs, and colors represent the status of each job. In this case, failures didn't prevent subsequent jobs to be completed.

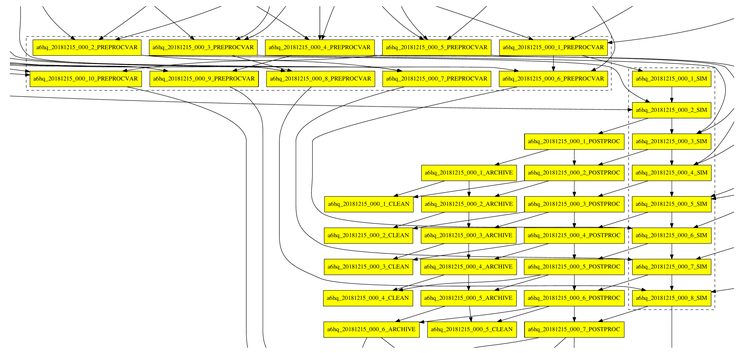
This feature is used in some of the operational forecast executions performed at BSC-ES, for instance in the Copernicus Atmosphere Monitoring Service (CAMS) air quality ensemble forecast executions done with the in-house MONARCH model. In this case, some contingency checks were added to ensure that the forecast is produced even if some of the input data are not available.

Feature example 2: Multi wrappers

Queuing times in HPC environments can be very long for many different reasons. To overcome this, Autosubmit includes the Wrapper feature which packs multiple tasks together in a single job. Instead of submitting N individual jobs separately, only one is submitted requesting the total amount of resources needed by all of them (the sum of all wallclock times and/or number of processors), thus reducing the total queuing time.

Different types of wrappers can be created depending on the type of workflow to be run: a long simulation with many sequential jobs, an ensemble with independent jobs that can be run in parallel, etc.

The first implementation of this Autosubmit feature only allowed one type of wrapper to be used in a given experiment. With the emergence of more complex workflows, it became necessary to use different wrappers for different kinds of jobs in the same workflow.



A fragment of a workflow with wrapped jobs (dashed lines). Each wrapper is submitted as a single job and Autosubmit takes care of executing inner tasks individually taking into account the dependencies. In this case, two different types of wrappers were used.

Conclusions

- Having a workflow manager fully developed at the BSC-ES has led to the adoption of an agile co-design approach as a result of the synergy between users and developers.
- This continuous interaction at all levels has resulted in an efficient and highly adaptable framework, perfectly aligned with the constantly evolving user needs.
- This strategy has had a very positive impact on different operational and research projects, demonstrating how it can help to achieve high scientific productivity.

References

– D. Manubens-Gil, J. Vegas-Regidor, C. Prodhomme, O. Mula-Valls and F. J. Doblas-Reyes, "Seamless management of ensemble climate prediction experiments on HPC platforms," 2016 International Conference on High Performance Computing & Simulation (HPCS), Innsbruck, 2016, pp. 895-900. doi: 10.1109/HPCSim.2016.7568429

Acknowledgements

The research leading to these results has received funding from the EU HE Framework Programme under grant agreement n° GA 101056783